# CIFO এক্সটেন্ডেড প্রকল্পের جامع চূড়ান্ত প্রতিবেদন

**CIFO টিম**

**তারিখ:** ১৫ মে, ২০২৫

## নির্বাহী সারসংক্ষেপ

এই চূড়ান্ত প্রতিবেদনটি CIFO এক্সটেন্ডেড প্রকল্পের সমস্ত ধাপকে একত্রিত করে, যার লক্ষ্য ছিল মেটাহিউরিস্টিক অ্যালগরিদমের প্রয়োগ এবং বিশ্লেষণের মাধ্যমে ক্রীড়া দলের নির্বাচনকে অপ্টিমাইজ করা। প্রকল্পটি একক-প্রসেসর বাস্তবায়ন থেকে শুরু করে কোড অপ্টিমাইজেশান, বিভিন্ন সংখ্যক রানের সাথে মাল্টিপ্রসেসিংয়ের অন্বেষণ, এবং সবচেয়ে প্রতিশ্রুতিশীল অ্যালগরিদমগুলির জন্য প্যারামিটার পরিবর্তনের গভীর বিশ্লেষণ পর্যন্ত বিকশিত হয়েছে। প্রধান উপসংহারগুলি হিল ক্লাইম্বিংকে (বিশেষত ৫০০ পুনরাবৃত্তিসহ) সমাধানের গুণমান, সামঞ্জস্য এবং সম্পাদনের সময়ের মধ্যে ভারসাম্য রক্ষার ক্ষেত্রে সবচেয়ে কার্যকর পদ্ধতি হিসাবে নির্দেশ করে। যেসব ক্ষেত্রে নিখুঁত সামঞ্জস্যের সাথে সর্বোত্তম সম্ভাব্য ফিটনেসের নিশ্চয়তা প্রয়োজন, সেখানে জেনেটিক অ্যালগরিদমের একটি অপ্টিমাইজড কনফিগারেশন (GA\_Config\_4, ০.১৫ মিউটেশন হার সহ) উচ্চতর প্রমাণিত হয়েছে, যদিও এতে গণনামূলক ব্যয় বেশি। এই নথিটি প্রতিটি ধাপের পদ্ধতি, ফলাফল এবং উপসংহার বিস্তারিতভাবে বর্ণনা করে, ভবিষ্যতের কাজের জন্য সুপারিশ এবং ব্যবহৃত বিশ্লেষণ প্রক্রিয়ার একটি চিত্রসহ সমাপ্তি টানা হয়েছে।

## ১. প্রকল্পের সাধারণ ভূমিকা

CIFO এক্সটেন্ডেড প্রকল্পের কেন্দ্রীয় চ্যালেঞ্জ ছিল উপলব্ধ খেলোয়াড়দের একটি পুল থেকে একাধিক ক্রীড়া দল গঠনের জটিল সমস্যার জন্য কার্যকর গণনামূলক পদ্ধতি তৈরি এবং মূল্যায়ন করা। এই কম্বিনেটোরিয়াল অপ্টিমাইজেশান সমস্যাটি বিভিন্ন সীমাবদ্ধতা দ্বারা চিহ্নিত, যার মধ্যে রয়েছে দলের বাজেট, প্রতি দলে খেলোয়াড়ের সংখ্যা এবং নির্দিষ্ট অবস্থানগত প্রয়োজনীয়তা। অপ্টিমাইজেশানের প্রাথমিক উদ্দেশ্য ছিল গঠিত দলগুলির মধ্যে গড় দক্ষতার বৈচিত্র্য হ্রাস করা, যার ফলে প্রতিযোগিতামূলক ভারসাম্য বৃদ্ধি পায়।

প্রকল্প জুড়ে, হিল ক্লাইম্বিং (HC), সিমুলেটেড অ্যানিলিং (SA), এবং বিভিন্ন অপারেটরসহ জেনেটিক অ্যালগরিদম (GAs) সহ বিভিন্ন মেটাহিউরিস্টিক অ্যালগরিদম অন্বেষণ করা হয়েছে। অনুসন্ধান বিভিন্ন স্বতন্ত্র ধাপের মাধ্যমে এগিয়েছে:

১. **একক-প্রসেসর পর্যায়:** অ্যালগরিদমগুলির প্রাথমিক বাস্তবায়ন, বাধা সনাক্তকরণের জন্য প্রোফাইলিং এবং কোড অপ্টিমাইজেশান (deepcopy হ্রাস এবং ভেক্টরাইজেশন)। ২. **মাল্টিপ্রসেসিং পর্যায় - ৫ রান:** পরিসংখ্যানগতভাবে আরও শক্তিশালী ডেটা পেতে এবং গভীর বিশ্লেষণের জন্য প্রার্থী নির্বাচন করতে অ্যালগরিদমগুলির সমান্তরালভাবে প্রাথমিক কর্মক্ষমতা মূল্যায়ন। ৩. **মাল্টিপ্রসেসিং পর্যায় - প্রতিশ্রুতিশীল অ্যালগরিদমগুলির সাথে ৩০ রান:** নির্বাচিত অ্যালগরিদমগুলির (HC, SA, এবং দুটি GA কনফিগারেশন) তাদের বেস প্যারামিটারগুলির সাথে আরও কঠোর পরিসংখ্যানগত বিশ্লেষণ। ৪. **প্যারামিটার পরিবর্তন পর্যায় - প্রতি পরিবর্তনে ৩০ রান:** বিভিন্ন হাইপারপ্যারামিটারের (HC-এর জন্য পুনরাবৃত্তির সংখ্যা; নির্বাচিত GA-গুলির জন্য মিউটেশন হার, جمعیت আকার, প্রজন্মের সংখ্যা) কর্মক্ষমতার উপর প্রভাবের বিস্তারিত অনুসন্ধান।

এই প্রতিবেদনের লক্ষ্য হল প্রাথমিক ধারণা থেকে চূড়ান্ত উপসংহার এবং সুপারিশ পর্যন্ত সমগ্র প্রক্রিয়ার একটি একত্রিত দৃশ্য উপস্থাপন করা।

## ২. একক-প্রসেসর পর্যায়: প্রাথমিক বাস্তবায়ন এবং অপ্টিমাইজেশান

### ২.১. একক-প্রসেসর কোড আর্কিটেকচার (প্রাথমিক সংস্করণ)

একক-প্রসেসর পর্যায়ের জন্য কোড আর্কিটেকচারটি মডুলারভাবে ডিজাইন করা হয়েছিল যাতে মেটাহিউরিস্টিক অ্যালগরিদমগুলির বিভিন্ন উপাদানগুলির বাস্তবায়ন, পরীক্ষা এবং প্রতিস্থাপন সহজতর হয়। এই আর্কিটেকচারে জড়িত প্রধান পাইথন ফাইলগুলি হল:

* solution.py: সমস্যার সমাধানের উপস্থাপনা (খেলোয়াড়দের দলে নিয়োগ) এবং এর বৈধতা ও গুণমান (ফিটনেস) মূল্যায়নের পদ্ধতি সংজ্ঞায়িত করে। এটি প্রতিবেশী সমাধান তৈরি বা বিদ্যমান সমাধানগুলি পরিবর্তন করার পদ্ধতিও অন্তর্ভুক্ত করে।
* evolution.py: প্রধান মেটাহিউরিস্টিক অ্যালগরিদমগুলির বাস্তবায়ন ধারণ করে: হিল ক্লাইম্বিং, সিমুলেটেড অ্যানিলিং এবং জেনেরিক জেনেটিক অ্যালগরিদম।
* operators.py: জেনেটিক অ্যালগরিদম দ্বারা ব্যবহৃত বিভিন্ন জেনেটিক অপারেটর যেমন নির্বাচন, ক্রসওভার এবং মিউটেশন অপারেটরগুলি বাস্তবায়ন করে।
* main\_script\_sp.py: প্রধান স্ক্রিপ্ট যা অ্যালগরিদমগুলির সম্পাদনকে অর্কেস্ট্রেট করে, খেলোয়াড়ের ডেটা লোড করে, অ্যালগরিদম প্যারামিটারগুলি কনফিগার করে এবং ফলাফল সংগ্রহ/উপস্থাপন করে।

#### ২.১.১. সমাধান উপস্থাপনা এবং ডেটা কাঠামো

সমাধান উপস্থাপনা এবং ডেটা কাঠামো পছন্দের পিছনে মূল কারণ ছিল প্রয়োজনীয় ক্রিয়াকলাপগুলির জন্য সরলতা এবং দক্ষতা।

* **সমাধান উপস্থাপনা (LeagueSolution solution.py তে):** একটি সমাধান একটি তালিকা (বা পরে, একটি NumPy অ্যারে) হিসাবে উপস্থাপিত হয় যেখানে সূচকটি একজন খেলোয়াড়ের আইডির সাথে সঙ্গতিপূর্ণ এবং সেই সূচকের মানটি সেই খেলোয়াড়কে যে দলে নিয়োগ করা হয়েছে তার আইডির সাথে সঙ্গতিপূর্ণ। উদাহরণস্বরূপ, assignment[player\_id] = team\_id।
  + **মূল কারণ:** এই উপস্থাপনাটি সহজবোধ্য, পরিচালনা করা সহজ এবং যেকোনো খেলোয়াড়ের দলে দ্রুত অ্যাক্সেসের অনুমতি দেয়। এটি সংক্ষিপ্তও।
* **খেলোয়াড়ের ডেটা:** খেলোয়াড়ের ডেটা (আইডি, নাম, অবস্থান, বেতন, দক্ষতা) প্রাথমিকভাবে একটি CSV ফাইল থেকে প্রধান স্ক্রিপ্টে একটি সুবিধাজনক ডেটা কাঠামোতে (যেমন অভিধানের তালিকা বা একটি পান্ডাস ডেটাফ্রেম) লোড করা হয়। সমাধান এবং অ্যালগরিদম ক্লাসগুলির অভ্যন্তরীণ ক্রিয়াকলাপগুলির জন্য, এই ডেটা প্রায়শই গণনার জন্য আরও অপ্টিমাইজ করা ফর্ম্যাটে পাস বা রূপান্তরিত হয় (যেমন বেতন, দক্ষতা এবং সংখ্যাসূচক অবস্থানগুলির জন্য NumPy অ্যারে, যেমন ভেক্টরাইজেশন পর্যায়ে করা হয়েছিল)।
  + **মূল কারণ:** CSV ডেটা ইনপুটের জন্য একটি সাধারণ এবং সহজেই ব্যবহারযোগ্য ফর্ম্যাট। অভ্যন্তরীণভাবে, NumPy অ্যারেতে রূপান্তর (বিশেষত ভেক্টরাইজেশনের পরে) সংখ্যাসূচক গণনা দ্রুত করার লক্ষ্য রাখে।
* **দলের কাঠামো:** অন্তর্নিহিতভাবে, দলগুলি খেলোয়াড়দের সংগ্রহ। সমাধান উপস্থাপনা প্রতিটি দলের গঠন সহজে পুনর্গঠন করার অনুমতি দেয়, তাদের নির্ধারিত team\_id এর উপর ভিত্তি করে খেলোয়াড়দের ফিল্টার করে।

#### ২.১.২. বাস্তবায়িত অ্যালগরিদম (প্রাথমিক, অপ্টিমাইজ না করা সংস্করণ)

এই প্রাথমিক পর্যায়ের জন্য তিন ধরনের প্রধান মেটাহিউরিস্টিক অ্যালগরিদম নির্বাচন করা হয়েছিল, তাদের জনপ্রিয়তা এবং সমাধান স্থান অনুসন্ধানের বিভিন্ন পদ্ধতির কারণে:

১. **হিল ক্লাইম্বিং (HC):** \* **বর্ণনা:** একটি পুনরাবৃত্তিমূলক স্থানীয় অনুসন্ধান অ্যালগরিদম যা একটি নির্বিচারে সমাধান দিয়ে শুরু হয় এবং আরও ভাল ফিটনেস সহ প্রতিবেশীর দিকে অগ্রসর হয়ে আরও ভাল সমাধান খুঁজে বের করার চেষ্টা করে। এটি যখন একটি স্থানীয় সর্বোত্তমে পৌঁছায় তখন এটি শেষ হয়, যেখানে কোনও প্রতিবেশীর আরও ভাল ফিটনেস নেই। \* **নির্বাচনের মূল কারণ:** এটি বাস্তবায়ন এবং বোঝার জন্য সবচেয়ে সহজ অপ্টিমাইজেশান অ্যালগরিদমগুলির মধ্যে একটি। এটি আরও জটিল অ্যালগরিদমগুলির সাথে তুলনা করার জন্য একটি ভাল ভিত্তি হিসাবে কাজ করে। এটি দ্রুত কিন্তু স্থানীয় সর্বোত্তমগুলিতে আটকে যাওয়ার প্রবণতা রয়েছে।

২. **সিমুলেটেড অ্যানিলিং (SA):** \* **বর্ণনা:** ধাতুবিদ্যায় অ্যানিলিং প্রক্রিয়া থেকে অনুপ্রাণিত একটি সম্ভাব্যতা অপ্টিমাইজেশান কৌশল। এটি একটি নির্দিষ্ট সম্ভাবনার সাথে আরও খারাপ সমাধানের দিকে অগ্রসর হওয়ার অনুমতি দেয়, যা অ্যালগরিদম অগ্রসর হওয়ার সাথে সাথে হ্রাস পায় (“তাপমাত্রা” কমে যায়)। এটি স্থানীয় সর্বোত্তমগুলি থেকে পালাতে সহায়তা করে। \* **নির্বাচনের মূল কারণ:** এটি হিল ক্লাইম্বিংয়ের চেয়ে ভাল বিশ্বব্যাপী অনুসন্ধান ক্ষমতা সরবরাহ করে, স্থানীয় সর্বোত্তমগুলি এড়ানোর ক্ষমতা সহ। এর কার্যকারিতা কুলিং সময়সূচীর সঠিক প্যারামিটারাইজেশনের উপর নির্ভর করে।

৩. **জেনেটিক অ্যালগরিদম (GAs):** \* **বর্ণনা:** জৈবিক বিবর্তন থেকে অনুপ্রাণিত অনুসন্ধান অ্যালগরিদম। তারা প্রার্থী সমাধানগুলির একটি জনসংখ্যা বজায় রাখে যা নির্বাচন, ক্রসওভার (পুনর্মিলন) এবং মিউটেশনের মতো জেনেটিক অপারেটরগুলির মাধ্যমে প্রজন্মের পর প্রজন্ম ধরে বিকশিত হয়। \* **নির্বাচনের মূল কারণ:** এগুলি জটিল অপ্টিমাইজেশান সমস্যার জন্য শক্তিশালী এবং বৃহৎ সমাধান স্থানগুলি কার্যকরভাবে অন্বেষণ করতে পারে। তাদের জনসংখ্যা-ভিত্তিক প্রকৃতি বৈচিত্র্য বজায় রাখতে এবং অনুসন্ধান স্থানের একাধিক অঞ্চল একযোগে অন্বেষণ করতে দেয়।

#### ২.১.৩. পরীক্ষার জন্য অপারেটর নির্বাচন (জেনেটিক অ্যালগরিদম)

জেনেটিক অ্যালগরিদমগুলির জন্য, বিভিন্ন বিবর্তনীয় কৌশল পরীক্ষা করার অনুমতি দেওয়ার জন্য বিভিন্ন অপারেটর প্রয়োগ করা হয়েছিল:

* **নির্বাচন অপারেটর:** বর্তমান জনসংখ্যা থেকে কোন ব্যক্তিরা প্রজননের জন্য নির্বাচিত হয় তা নির্ধারণ করে।
  + selection\_tournament: টুর্নামেন্ট নির্বাচন।
  + selection\_roulette\_wheel: রুলেট হুইল নির্বাচন।
  + selection\_ranking: র‍্যাঙ্ক-ভিত্তিক নির্বাচন।
  + selection\_boltzmann।
* **ক্রসওভার অপারেটর:** দুই পিতামাতার জেনেটিক উপাদান একত্রিত করে এক বা একাধিক সন্তান তৈরি করে।
  + crossover\_one\_point: এক-বিন্দু ক্রসওভার।
  + crossover\_uniform: অভিন্ন ক্রসওভার।
  + \_prefer\_valid সংস্করণগুলি আরও সরাসরি বৈধ সন্তান তৈরি করার চেষ্টা করার জন্য।
* **মিউটেশন অপারেটর:** ব্যক্তিদের মধ্যে ছোট এলোমেলো পরিবর্তন প্রবর্তন করে।
  + mutation\_random\_player\_team\_change: এলোমেলোভাবে একজন খেলোয়াড়ের দল পরিবর্তন করে।
  + mutation\_swap\_players\_between\_teams: দুটি দলের মধ্যে দুজন খেলোয়াড় অদলবদল করে।

এই অ্যালগরিদম এবং অপারেটরগুলির প্রাথমিক নির্বাচন মেটাহিউরিস্টিক পদ্ধতির একটি বর্ণালী কভার করার লক্ষ্য ছিল।

### ২.২. প্রাথমিক অপ্টিমাইজেশানের ফলাফল (একক-প্রসেসর)

প্রাথমিক বাস্তবায়নের পরে, একটি প্রোফাইলিং এবং অপ্টিমাইজেশান পর্যায় চালানো হয়েছিল। দুটি প্রধান অপ্টিমাইজেশান ক্ষেত্র deepcopy এর ব্যবহার হ্রাস এবং জটিল ফাংশনগুলির ভেক্টরাইজেশনের উপর দৃষ্টি নিবদ্ধ করে।

#### ২.২.১. প্রাথমিক প্রোফাইলিং এবং বাধা সনাক্তকরণ

স্ক্রিপ্টের (main\_script\_sp\_timing\_estimate.py) প্রাথমিক সম্পাদন প্রকাশ করেছে যে মোট সময় ছিল প্রায় **৮৫.৬৬ সেকেন্ড**। cProfile হাইলাইট করেছে:

* **সিমুলেটেড অ্যানিলিং (SA):** সবচেয়ে ধীর, প্রায় **৭৩.১৭ সেকেন্ড** ব্যয় করে।
* **copy.deepcopy:** ধীরগতির প্রধান কারণ, ২৩ মিলিয়নেরও বেশি বার কল করা হয়েছে, মোট **৫৩.৪৯ সেকেন্ড**।
* **সমাধান ফাংশন:** get\_random\_neighbor, fitness, এবং is\_valid এছাড়াও উল্লেখযোগ্য সময় ব্যয় করেছে।

#### ২.২.২. সিমুলেটেড অ্যানিলিং-এ deepcopy ব্যবহারের অপ্টিমাইজেশান

* **পরিবর্তন:** SA-তে, current\_solution আপডেট করার জন্য deepcopy(neighbor\_solution)-এর কলগুলি সরাসরি অ্যাসাইনমেন্ট দ্বারা প্রতিস্থাপিত হয়েছিল। best\_solution-এর জন্য deepcopy কল বজায় রাখা হয়েছিল।
* **প্রভাব:** মোট স্ক্রিপ্ট সময় **৪১.৪২ সেকেন্ডে** হ্রাস (প্রায় ৫১.৭% উন্নতি)। SA সময় **২৮.৯১ সেকেন্ডে** হ্রাস (প্রায় ৬০.৫% উন্নতি)।

#### ২.২.৩. is\_valid() এবং fitness() ফাংশনগুলির ভেক্টরাইজেশন

* **পরিবর্তন:** solution.py-তে is\_valid() এবং fitness() ফাংশনগুলি NumPy ব্যবহার করে পুনরায় লেখা হয়েছিল।
* **প্রভাব:** সমস্ত অপ্টিমাইজেশান সহ মোট স্ক্রিপ্ট সময় ছিল প্রায় **৪০.৯৬ সেকেন্ড**। ভেক্টরাইজেশন এই প্রসঙ্গে বিশ্বব্যাপী সময়ে উল্লেখযোগ্য অতিরিক্ত উন্নতি আনেনি তবে কোডটিকে আরও শক্তিশালী করেছে।

### ২.৩. একক-প্রসেসর পর্যায়ের উপসংহার

একক-প্রসেসর পর্যায়টি অত্যন্ত গুরুত্বপূর্ণ ছিল। deepcopy অপ্টিমাইজ করার ফলে সম্পাদনের সময় উল্লেখযোগ্যভাবে হ্রাস পেয়েছে। ভেক্টরাইজেশন কোডবেসকে আধুনিকীকরণ করেছে, এটিকে আরও চাহিদাপূর্ণ পরিস্থিতির জন্য প্রস্তুত করেছে। এই ফলাফলগুলি মাল্টিপ্রসেসিং অন্বেষণের জন্য মূল্যবান তথ্য সরবরাহ করেছে।

## ৩. মাল্টিপ্রসেসিং পর্যায় - ৫ রান: প্রাথমিক মূল্যায়ন

এই প্রকল্পের এই অংশে অপ্টিমাইজেশান অ্যালগরিদমগুলির (হিল ক্লাইম্বিং, সিমুলেটেড অ্যানিলিং, এবং জেনেটিক অ্যালগরিদমগুলির চারটি কনফিগারেশন) কর্মক্ষমতা মূল্যায়ন করা হয়েছে যখন সেগুলি সমান্তরালভাবে একাধিকবার চালানো হয়। এই পর্যায়ের প্রধান উদ্দেশ্য ছিল প্রতিটি পদ্ধতির কার্যকারিতা এবং দক্ষতার উপর পরিসংখ্যানগতভাবে আরও শক্তিশালী ডেটা প্রাপ্ত করা, প্রতিটি অ্যালগরিদম ৫ বার চালিয়ে। মাল্টিপ্রসেসিংয়ের ব্যবহার এই সমবর্তী সম্পাদনগুলিকে অনুমতি দিয়েছে, মোট পরীক্ষামূলক সময়কে অপ্টিমাইজ করেছে।

### ৩.১. পরীক্ষামূলক পদ্ধতি

মূল্যায়িত অ্যালগরিদমগুলি ছিল: হিল ক্লাইম্বিং (HC), সিমুলেটেড অ্যানিলিং (SA), এবং জেনেটিক অ্যালগরিদমগুলির চারটি কনফিগারেশন (GA\_Config\_1, GA\_Config\_2, GA\_Config\_3, GA\_Config\_4), যেখানে মিউটেশন, ক্রসওভার এবং নির্বাচন অপারেটরগুলি পরিবর্তিত হয়েছে। প্রতিটি অ্যালগরিদম/কনফিগারেশন সমান্তরালভাবে ৫ বার চালানো হয়েছিল। সংগৃহীত মেট্রিকগুলির মধ্যে ছিল সেরা সামগ্রিক ফিটনেস, গড় ফিটনেস, ফিটনেসের স্ট্যান্ডার্ড ডেভিয়েশন এবং গড় সম্পাদনের সময়।

### ৩.২. ফলাফল এবং আলোচনা (৫ রান)

প্রতিটি অ্যালগরিদমের ৫টি রানের জন্য main\_script\_mp.py স্ক্রিপ্টটি চালানোর পরে, ফলাফলগুলি ছিল:

* **সেরা সামগ্রিক ফিটনেস:** সমস্ত অ্যালগরিদম (SA, GA\_Config\_1, GA\_Config\_2, GA\_Config\_3, এবং GA\_Config\_4) প্রায় **০.০৫৭১৪৩** এর একই সেরা সামগ্রিক ফিটনেস মান অর্জন করতে সক্ষম হয়েছিল। হিল ক্লাইম্বিং, যদিও এই মানে পৌঁছেছে, তার গড় ফিটনেস কিছুটা বেশি ছিল (০.০৬৭১)।
* **সামঞ্জস্য (ফিটনেসের স্ট্যান্ডার্ড ডেভিয়েশন):** সিমুলেটেড অ্যানিলিং এবং GA\_Config\_4 ফিটনেসের স্ট্যান্ডার্ড ডেভিয়েশন **০.০** সহ দাঁড়িয়েছে, যা নির্দেশ করে যে তারা সমস্ত ৫টি রানে একই সর্বোত্তম মানে পৌঁছেছে।
* **দক্ষতা (গড় সম্পাদনের সময়):**
  + হিল ক্লাইম্বিং: ~০.৪৭ সেকেন্ড (দ্রুততম)।
  + GA\_Config\_1: ~৬.০৫ সেকেন্ড।
  + GA\_Config\_4: ~৮.১৯ সেকেন্ড।
  + সিমুলেটেড অ্যানিলিং: ~১৭.৬০ সেকেন্ড।
  + GA\_Config\_2 এবং GA\_Config\_3: ~৬১-৬২ সেকেন্ড (সবচেয়ে ধীর)।

### ৩.৩. প্রাথমিক উপসংহার (৫ রান)

৫টি রানের ফলাফলের উপর ভিত্তি করে:

* **হিল ক্লাইম্বিং** অত্যন্ত দ্রুত প্রমাণিত হয়েছে।
* **সিমুলেটেড অ্যানিলিং** ধারাবাহিকভাবে সেরা সমাধান খুঁজে পেয়েছে, তবে মাঝারি সময়ের খরচে।
* **GA\_Config\_4** একটি শক্তিশালী প্রার্থী হিসাবে আবির্ভূত হয়েছে, সেরা ফিটনেস এবং সামঞ্জস্যের ক্ষেত্রে SA-এর সাথে মিলেছে, তবে দ্রুততর হচ্ছে।
* **GA\_Config\_1** GA-গুলির মধ্যে তার গতির জন্য উল্লেখযোগ্য ছিল, যদিও কম সামঞ্জস্য সহ।

এই ফলাফলগুলি ৩০-রান পর্যায়ের জন্য অ্যালগরিদম নির্বাচনের ভিত্তি সরবরাহ করেছে।

## ৪. মাল্টিপ্রসেসিং পর্যায় - প্রতিশ্রুতিশীল অ্যালগরিদমগুলির সাথে ৩০ রান: পরিসংখ্যানগত বিশ্লেষণ

এই প্রকল্পের এই অংশে পূর্ববর্তী ৫-রান পর্যায়ে সবচেয়ে প্রতিশ্রুতিশীল হিসাবে চিহ্নিত চারটি অ্যালগরিদমের (হিল ক্লাইম্বিং, সিমুলেটেড অ্যানিলিং, GA\_Config\_1, এবং GA\_Config\_4) একটি বিস্তারিত পরিসংখ্যানগত বিশ্লেষণ উপস্থাপন করা হয়েছে। পরিসংখ্যানগতভাবে আরও শক্তিশালী ফলাফল পেতে প্রতিটি অ্যালগরিদম সমান্তরালভাবে ৩০ বার চালানো হয়েছিল।

### ৪.১. পদ্ধতি

অ্যালগরিদমগুলি তাদের বেস প্যারামিটারগুলির সাথে চালানো হয়েছিল (৫-রান পর্যায়ের মতোই)। বিশ্লেষিত মেট্রিকগুলির মধ্যে ছিল গড় ফিটনেস, ফিটনেসের স্ট্যান্ডার্ড ডেভিয়েশন, সেরা সামগ্রিক ফিটনেস এবং গড় সম্পাদনের সময়।

### ৪.২. ফলাফল এবং পরিসংখ্যানগত বিশ্লেষণ (প্রাথমিক ৩০ রান)

* **সেরা সামগ্রিক ফিটনেস:** চারটি অ্যালগরিদমই (HC, SA, GA\_Config\_1, GA\_Config\_4) প্রায় **০.০৫৭১** এর একই সেরা সামগ্রিক ফিটনেস মান অর্জন করতে সক্ষম হয়েছিল।
* **গড় ফিটনেস এবং সামঞ্জস্য:**
  + HC, SA, এবং GA\_Config\_4 একই গড় ফিটনেস (০.০৬০৫) এবং একই স্ট্যান্ডার্ড ডেভিয়েশন (০.০১২৪) দেখিয়েছিল।
  + GA\_Config\_1 গড় ফিটনেসে কিছুটা খারাপ পারফর্ম করেছে (০.০৬৮৮) এবং এর পরিবর্তনশীলতা বেশি ছিল (স্ট্যান্ডার্ড ডেভিয়েশন ০.০২১০)।
* **দক্ষতা (গড় সম্পাদনের সময়):**
  + হিল ক্লাইম্বিং: ~০.৬৩ সেকেন্ড (দ্রুততম)।
  + GA\_Config\_1: ~৫.৯৭ সেকেন্ড।
  + GA\_Config\_4: ~৮.৩০ সেকেন্ড।
  + সিমুলেটেড অ্যানিলিং: ~১৮.২১ সেকেন্ড (সবচেয়ে ধীর)।

### ৪.৩. উপসংহার (প্রাথমিক ৩০ রান)

* **হিল ক্লাইম্বিং** সবচেয়ে দক্ষ অ্যালগরিদম হিসাবে আবির্ভূত হয়েছে, SA এবং GA\_Config\_4 এর মতো একই সমাধানের গুণমান অর্জন করেছে, তবে সময়ের একটি ভগ্নাংশে।
* HC, SA, এবং GA\_Config\_4 সমানভাবে সামঞ্জস্যপূর্ণ ছিল।
* HC গুণমান, সামঞ্জস্য এবং দক্ষতার মধ্যে সেরা ভারসাম্য সরবরাহ করেছে।

এই বিশ্লেষণের ফলে চূড়ান্ত প্যারামিটার পরিবর্তন পর্যায়টি হিল ক্লাইম্বিং এবং দুটি সবচেয়ে প্রতিশ্রুতিশীল GA কনফিগারেশনের (GA\_Config\_1 এবং GA\_Config\_4) উপর দৃষ্টি নিবদ্ধ করার সিদ্ধান্ত নেওয়া হয়েছিল, SA-কে তার তুলনামূলক ধীরগতির কারণে বাদ দেওয়া হয়েছিল কারণ HC বা অপ্টিমাইজ করা GA\_Config\_4 এর উপর কোনও গুণগত সুবিধা ছিল না।

## ৫. প্যারামিটার পরিবর্তন পর্যায় - প্রতি পরিবর্তনে ৩০ রান: বিস্তারিত পরিসংখ্যানগত বিশ্লেষণ

এই বিভাগে হিল ক্লাইম্বিং (HC) এবং দুটি জেনেটিক অ্যালগরিদম কনফিগারেশন (GA\_Config\_1 এবং GA\_Config\_4) এ প্রয়োগ করা ১৯টি প্যারামিটার পরিবর্তনের প্রতিটির জন্য ৩০টি পুনরাবৃত্তি সম্পাদনের পরে প্রাপ্ত ফলাফলগুলির পরিসংখ্যানগত বিশ্লেষণ বিস্তারিতভাবে বর্ণনা করা হয়েছে। এই পর্যায়ের উদ্দেশ্য ছিল প্রতিটি অ্যালগরিদমের জন্য সবচেয়ে কার্যকর হাইপারপ্যারামিটার সেটিংস সনাক্ত করা।

### ৫.১. প্যারামিটার পরিবর্তনের জন্য জেনেটিক অ্যালগরিদম নির্বাচনের যৌক্তিকতা

GA\_Config\_1\_SwapConst1PtPreferVTournVarK এবং GA\_Config\_4\_TargetExchUnifPreferVTournVarK\_k5 কনফিগারেশনগুলি পূর্ববর্তী ৩০টি রানের (বেস প্যারামিটার সহ) রাউন্ডে তাদের প্রতিশ্রুতিশীল কর্মক্ষমতার উপর ভিত্তি করে বিস্তারিত প্যারামিটার পরিবর্তন পর্যায়ের জন্য নির্বাচিত হয়েছিল।

* GA\_Config\_1 ছিল সেই পূর্ববর্তী পর্যায়ে দ্রুততম GA কনফিগারেশন যা সেরা ফিটনেস অর্জনের ক্ষমতাও প্রদর্শন করেছিল, যদিও অন্যদের তুলনায় কম সামঞ্জস্য সহ। এর আর্কিটেকচার, mutate\_swap\_constrained এবং crossover\_one\_point\_prefer\_valid সহ, একটি আরও ক্লাসিক এবং কম বিঘ্নিত পদ্ধতির প্রতিনিধিত্ব করে, যার কর্মক্ষমতা সূক্ষ্ম-টিউন করা যেতে পারে।
* GA\_Config\_4 খুব ভাল সামঞ্জস্য এবং প্রতিযোগিতামূলক সম্পাদনের সময় সহ সেরা ফিটনেস অর্জনের জন্য দাঁড়িয়েছে। এর অপারেটরগুলির সংমিশ্রণ (লক্ষ্যযুক্ত মিউটেশন mutate\_targeted\_player\_exchange এবং অভিন্ন ক্রসওভার crossover\_uniform\_prefer\_valid) সমাধান স্থানের অন্বেষণ এবং শোষণের জন্য ভাল সম্ভাবনার পরামর্শ দিয়েছে, যা এর প্যারামিটারগুলির আরও গভীর তদন্তকে ন্যায্যতা দিয়েছে।

প্রত্যাশা ছিল যে মিউটেশন হার, জনসংখ্যা আকার এবং প্রজন্মের সংখ্যার মতো প্যারামিটারগুলি পরিবর্তন করে আমরা এই দুটি স্বতন্ত্র GA আর্কিটেকচারের কর্মক্ষমতা আরও পরিমার্জন করতে পারব।

### ৫.২. প্যারামিটার পরিবর্তন পদ্ধতি

* **হিল ক্লাইম্বিং (HC):** **সর্বাধিক পুনরাবৃত্তির সংখ্যা** পরিবর্তন করা হয়েছিল: [৫০০, ১০০০ (বেস), ১৫০০]।
* **GA\_Config\_1 এবং GA\_Config\_4:** প্রতিটির জন্য, নিম্নলিখিতগুলি পৃথকভাবে পরিবর্তন করা হয়েছিল (অন্যান্য বেস প্যারামিটারগুলি তাদের মূল মানগুলিতে রেখে: PopSize=50, NumGen=100, MutRate=0.1):
  + **মিউটেশন হার (MutRate):** [০.০৫, ০.১৫, ০.২৫]
  + **জনসংখ্যা আকার (PopSize):** [৩০, ৭৫]
  + **প্রজন্মের সংখ্যা (NumGen):** [৭৫, ১৫০] ফলস্বরূপ ১৯টি পরিবর্তনের প্রতিটি ৩০ বার চালানো হয়েছিল।

### ৫.৩. প্যারামিটার পরিবর্তনের ফলাফল এবং বিশ্লেষণ

**বিশ্বব্যাপী মূল পর্যবেক্ষণ:** সমস্ত ১৯টি পরীক্ষিত কনফিগারেশন, তাদের ৩০টি রানের জুড়ে, অন্তত একবার **০.০৫৭১** এর একই বিশ্বব্যাপী সেরা ফিটনেস খুঁজে পেতে সক্ষম হয়েছিল।

#### ৫.৩.১. অপ্টিমাইজড হিল ক্লাইম্বিং (HC)

* **HC\_Iter\_500 (সর্বাধিক পুনরাবৃত্তি = ৫০০):**
  + গড় ফিটনেস: ০.০৫৮৮, স্ট্যান্ডার্ড ডেভিয়েশন: ০.০০৮৯, গড় সময়: ০.৪১ সেকেন্ড।
* **HC-এর জন্য উপসংহার:** HC\_Iter\_500 কনফিগারেশনটি ছিল সবচেয়ে কার্যকর, সেরা গড় ফিটনেস, সর্বনিম্ন স্ট্যান্ডার্ড ডেভিয়েশন এবং অত্যন্ত কম সম্পাদনের সময় সহ। পুনরাবৃত্তি বৃদ্ধি কোনও উন্নতি আনেনি।

#### ৫.৩.২. অপ্টিমাইজড জেনেটিক অ্যালগরিদম - কনফিগারেশন ১ (GA\_Config\_1)

* **সেরা পরিবর্তনগুলি:**
  + GA\_Config\_1\_mutation\_rate\_0.25 (Pop=50, Gen=100): গড় ফিটনেস: ০.০৫৮৮, স্ট্যান্ডার্ড ডেভিয়েশন: ০.০০৮৯, সময়: ৬.১৩ সেকেন্ড।
  + GA\_Config\_1\_population\_size\_75 (MutRate=0.1, Gen=100): গড় ফিটনেস: ০.০৫৮৮, স্ট্যান্ডার্ড ডেভিয়েশন: ০.০০৮৯, সময়: ৯.০৮ সেকেন্ড।
* **GA\_Config\_1-এর জন্য উপসংহার:** উভয় পরিবর্তনই HC\_Iter\_500-এর গুণমান এবং সামঞ্জস্যের সাথে মিলেছে, তবে যথেষ্ট দীর্ঘ সম্পাদনের সময় সহ। ০.২৫ মিউটেশন হার সহ পরিবর্তনটি সময়-দক্ষ ছিল বেশি।

#### ৫.৩.৩. অপ্টিমাইজড জেনেটিক অ্যালগরিদম - কনফিগারেশন ৪ (GA\_Config\_4)

* **সেরা পরিবর্তনগুলি (ধারাবাহিকভাবে ০.০৫৭১ গড় ফিটনেস এবং ~০.০ স্ট্যান্ডার্ড ডেভিয়েশন অর্জন করেছে):**
  + GA\_Config\_4\_mutation\_rate\_0.15 (Pop=50, Gen=100): সময়: ৮.৫৫ সেকেন্ড।
  + GA\_Config\_4\_mutation\_rate\_0.25 (Pop=50, Gen=100): সময়: ৮.৯০ সেকেন্ড।
  + GA\_Config\_4\_population\_size\_75 (MutRate=0.1, Gen=100): সময়: ১২.৯১ সেকেন্ড।
  + GA\_Config\_4\_generations\_150 (MutRate=0.1, Pop=50): সময়: ১২.৪৪ সেকেন্ড।
* **GA\_Config\_4-এর জন্য উপসংহার:** অসাধারণ সম্ভাবনা দেখিয়েছে, বেশ কয়েকটি পরিবর্তন ধারাবাহিকভাবে শূন্য স্ট্যান্ডার্ড ডেভিয়েশন সহ সেরা গড় ফিটনেস ০.০৫৭১ অর্জন করেছে। GA\_Config\_4\_mutation\_rate\_0.15 এই শীর্ষ-স্তরের কনফিগারেশনগুলির মধ্যে সবচেয়ে সময়-দক্ষ ছিল।

### ৫.৪. প্যারামিটার পরিবর্তন থেকে সাধারণ উপসংহার

১. **অর্জনযোগ্য বিশ্বব্যাপী সেরা ফিটনেস:** ০.০৫৭১, সমস্ত পরিবর্তন দ্বারা পৌঁছেছে। ২. **অপ্টিমাইজড হিল ক্লাইম্বিং (HC\_Iter\_500):** চরম দক্ষতা (০.৪১ সেকেন্ড), চমৎকার গড় ফিটনেস (০.০৫৮৮) এবং ভাল সামঞ্জস্যের জন্য দাঁড়িয়েছে। ৩. **অপ্টিমাইজড GA\_Config\_1 (GA\_Config\_1\_mutation\_rate\_0.25):** অপ্টিমাইজড HC-এর গুণমানের সাথে মিলেছে, তবে ধীর (৬.১৩ সেকেন্ড)। ৪. **অপ্টিমাইজড GA\_Config\_4 (GA\_Config\_4\_mutation\_rate\_0.15):** ধারাবাহিকভাবে সেরা গড় ফিটনেস (০.০৫৭১) শূন্য স্ট্যান্ডার্ড ডেভিয়েশন সহ অর্জন করেছে, যারা এই স্তরের পরিপূর্ণতায় পৌঁছেছে তাদের মধ্যে দ্রুততম (৮.৫৫ সেকেন্ড) হচ্ছে।

**প্যারামিটার পরিবর্তনের পরে চূড়ান্ত অ্যালগরিদম এবং কনফিগারেশন সুপারিশ:**

* **চমৎকার গুণমান সহ সর্বাধিক দক্ষতার জন্য:** HC\_Iter\_500।
* **নিখুঁত সামঞ্জস্য সহ সেরা ফিটনেসের নিশ্চয়তার জন্য:** GA\_Config\_4\_mutation\_rate\_0.15।

## ৬. চূড়ান্ত উপসংহার এবং সুপারিশ

### ৬.১. ফলাফলের সংশ্লেষণ

এই প্রকল্পটি ক্রীড়া দলের নির্বাচন অপ্টিমাইজ করার সমস্যার জন্য মেটাহিউরিস্টিক অ্যালগরিদমগুলির প্রয়োগ অন্বেষণ করেছে, একক-প্রসেসর বাস্তবায়ন থেকে বিস্তারিত প্যারামিটার পরিবর্তন বিশ্লেষণ পর্যন্ত অগ্রসর হয়েছে। প্রধান ফলাফলগুলি নিম্নরূপ সংক্ষিপ্ত করা যেতে পারে:

১. **কোড অপ্টিমাইজেশান:** সিমুলেটেড অ্যানিলিং-এ deepcopy ব্যবহার হ্রাস করার ফলে মোট সম্পাদনের সময়ে প্রায় ৫১.৭% কর্মক্ষমতা উন্নতি হয়েছে।

২. **মাল্টিপ্রসেসিং - ৫ রান:** সমস্ত অ্যালগরিদম একই সেরা সামগ্রিক ফিটনেস (০.০৫৭১৪৩) অর্জন করতে সক্ষম হয়েছিল, তবে সামঞ্জস্য এবং সম্পাদনের সময়ে উল্লেখযোগ্য পার্থক্য ছিল। হিল ক্লাইম্বিং ছিল দ্রুততম (০.৪৭ সেকেন্ড), যেখানে সিমুলেটেড অ্যানিলিং এবং GA\_Config\_4 ছিল সবচেয়ে সামঞ্জস্যপূর্ণ।

৩. **মাল্টিপ্রসেসিং - ৩০ রান:** হিল ক্লাইম্বিং, সিমুলেটেড অ্যানিলিং এবং GA\_Config\_4 সমাধানের গুণমানের (গড় ফিটনেস ০.০৬০৫, স্ট্যান্ডার্ড ডেভিয়েশন ০.০১২৪) দিক থেকে পরিসংখ্যানগতভাবে সমতুল্য কর্মক্ষমতা দেখিয়েছে, তবে হিল ক্লাইম্বিং উল্লেখযোগ্যভাবে দ্রুততর ছিল (SA-এর জন্য ১৮.২১ সেকেন্ডের তুলনায় ০.৬৩ সেকেন্ড)।

৪. **প্যারামিটার পরিবর্তন - প্রতি পরিবর্তনে ৩০ রান:** - ৫০০ পুনরাবৃত্তি সহ হিল ক্লাইম্বিং (HC\_Iter\_500) সবচেয়ে দক্ষ কনফিগারেশন হিসাবে আবির্ভূত হয়েছে, চমৎকার গড় ফিটনেস (০.০৫৮৮), ভাল সামঞ্জস্য (স্ট্যান্ডার্ড ডেভিয়েশন ০.০০৮৯) এবং অত্যন্ত কম সময় (০.৪১ সেকেন্ড) সহ। - ০.১৫ মিউটেশন হার সহ GA\_Config\_4 (GA\_Config\_4\_mutation\_rate\_0.15) ধারাবাহিকভাবে সেরা গড় ফিটনেস (০.০৫৭১) শূন্য স্ট্যান্ডার্ড ডেভিয়েশন সহ অর্জন করেছে, এই স্তরের পরিপূর্ণতা নিশ্চিতকারী কনফিগারেশনগুলির মধ্যে দ্রুততম (৮.৫৫ সেকেন্ড) হচ্ছে।

### ৬.২. বাস্তব প্রয়োগের জন্য সুপারিশ

প্রাপ্ত ফলাফলের উপর ভিত্তি করে, আমরা সুপারিশ করছি:

১. **সময়-সীমাবদ্ধ পরিস্থিতিগুলির জন্য:** ৫০০ পুনরাবৃত্তি সহ হিল ক্লাইম্বিং (HC\_Iter\_500) ব্যবহার করুন, যা সমাধানের গুণমান এবং গণনামূলক দক্ষতার মধ্যে একটি চমৎকার ভারসাম্য সরবরাহ করে।

২. **নিশ্চিত সেরা সমাধানের প্রয়োজন এমন পরিস্থিতিগুলির জন্য:** ০.১৫ মিউটেশন হার সহ জেনেটিক অ্যালগরিদম GA\_Config\_4 (GA\_Config\_4\_mutation\_rate\_0.15) ব্যবহার করুন, যা ধারাবাহিকভাবে সেরা সম্ভাব্য ফিটনেসের নিশ্চয়তা দেয়, যদিও উচ্চতর গণনামূলক ব্যয় সহ।

৩. **বৃহৎ আকারের অ্যাপ্লিকেশনগুলির জন্য:** একটি হাইব্রিড পদ্ধতির বাস্তবায়ন বিবেচনা করুন, দ্রুত একটি ভাল সমাধান পেতে হিল ক্লাইম্বিং দিয়ে শুরু করুন, তারপরে নির্দিষ্ট ক্ষেত্রে যেখানে সর্বাধিক গুণমান গুরুত্বপূর্ণ সেখানে অপ্টিমাইজ করা GA\_Config\_4 দিয়ে পরিমার্জন করুন।

### ৬.৩. ভবিষ্যতের কাজের জন্য পরামর্শ

১. **অ্যালগরিদম হাইব্রিডাইজেশনের অন্বেষণ:** হিল ক্লাইম্বিংকে জেনেটিক অ্যালগরিদমগুলির সাথে একত্রিত করার তদন্ত করুন, সম্ভাব্যভাবে GA দ্বারা উত্পন্ন সমাধানগুলি পরিমার্জন করতে HC ব্যবহার করে।

২. **স্কেলেবিলিটি বিশ্লেষণ:** বৃহত্তর আকারের সমস্যাগুলিতে অপ্টিমাইজ করা অ্যালগরিদমগুলির কর্মক্ষমতা পরীক্ষা করুন, আরও দল বা আরও বেশি খেলোয়াড় সহ।

৩. **অভ্যন্তরীণ অ্যালগরিদম সমান্তরালকরণ:** একাধিক রানের জন্য মাল্টিপ্রসেসিংয়ের বাইরে, অভ্যন্তরীণ অ্যালগরিদম সমান্তরালকরণ অন্বেষণ করুন, বিশেষত জেনেটিক অ্যালগরিদমগুলির জন্য (ব্যক্তিদের সমান্তরাল মূল্যায়ন)।

৪. **মেশিন লার্নিং কৌশলগুলির অন্বেষণ:** সমস্যার বৈশিষ্ট্যগুলির উপর ভিত্তি করে ভাল প্যারামিটার কনফিগারেশনগুলির পূর্বাভাস দেওয়ার জন্য শেখার কৌশলগুলির ব্যবহার তদন্ত করুন।

## ৭. বিশ্লেষণ প্রক্রিয়া চিত্র

নীচের চিত্রটি এই প্রকল্পে অনুসরণ করা বিশ্লেষণ প্রক্রিয়াটি চিত্রিত করে, প্রাথমিক বাস্তবায়ন থেকে চূড়ান্ত সুপারিশ পর্যন্ত:

┌─────────────────────────┐  
│ প্রাথমিক বাস্তবায়ন │  
│ (একক-প্রসেসর) │  
└───────────┬─────────────┘  
 ▼  
┌─────────────────────────┐  
│ প্রোফাইলিং ও অপ্টিমাইজেশন│  
│ - deepcopy হ্রাস │  
│ - ভেক্টরাইজেশন │  
└───────────┬─────────────┘  
 ▼  
┌─────────────────────────┐  
│ মাল্টিপ্রসেসিং │  
│ (৫ রান) │  
│ - HC, SA, ৪ GA কনফিগ │  
└───────────┬─────────────┘  
 ▼  
┌─────────────────────────┐  
│ প্রতিশ্রুতিশীল অ্যালগরিদম │  
│ নির্বাচন │  
│ - HC, SA, GA1, GA4 │  
└───────────┬─────────────┘  
 ▼  
┌─────────────────────────┐  
│ পরিসংখ্যানগত বিশ্লেষণ │  
│ (৩০ রান) │  
└───────────┬─────────────┘  
 ▼  
┌─────────────────────────┐  
│ প্যারামিটার পরিবর্তনের জন্য│  
│ নির্বাচন │  
│ - HC, GA1, GA4 │  
└───────────┬─────────────┘  
 ▼  
┌─────────────────────────┐  
│ প্যারামিটার পরিবর্তন │  
│ (৩০ রান/পরিবর্তন) │  
│ - ১৯ কনফিগারেশন │  
└───────────┬─────────────┘  
 ▼  
┌─────────────────────────┐  
│ চূড়ান্ত বিশ্লেষণ ও │  
│ সুপারিশ │  
│ - HC\_Iter\_500 │  
│ - GA4\_MutRate\_0.15 │  
└─────────────────────────┘

এই পদ্ধতিগত প্রক্রিয়াটি অ্যালগরিদম এবং প্যারামিটার স্থানের একটি পদ্ধতিগত অন্বেষণের অনুমতি দিয়েছে, যা বাস্তব প্রয়োগের জন্য সুপ্রতিষ্ঠিত সুপারিশগুলিতে পরিণত হয়েছে।

## ৬. একত্রিত পরিমাণগত বিশ্লেষণ এবং চূড়ান্ত সুপারিশসমূহ

পূর্ববর্তী বিভাগগুলিতে উপস্থাপিত বিস্তারিত বিশ্লেষণগুলির পরিপূরক হিসাবে, **CIFO এক্সটেন্ডেড প্রকল্পের অ্যালগরিদমগুলির বিস্তারিত পরিমাণগত বিশ্লেষণ** শিরোনামে একটি পৃথক নথি তৈরি করা হয়েছে। এই নথিটি (quantitative\_analysis/bn/quantitative\_analysis\_bn.md) সমস্ত পরীক্ষামূলক পর্যায় জুড়ে অ্যালগরিদমগুলির কার্যকারিতা সংখ্যাগতভাবে একত্রিত করে এবং তুলনা করে, গড় ফিটনেস, ফিটনেস স্ট্যান্ডার্ড ডেভিয়েশন, সামগ্রিক সেরা ফিটনেস এবং গড় সম্পাদনের সময়ের মতো মূল মেট্রিকগুলিতে আলোকপাত করে। এই নথিতে থাকা তুলনামূলক সারণী এবং গভীর আলোচনাগুলি নিম্নলিখিত উপসংহার এবং সুপারিশগুলির জন্য সংখ্যাগত ভিত্তি সরবরাহ করে।

### ৬.১. পরিমাণগত ফলাফলের সংশ্লেষণ

উপরে উল্লিখিত নথিতে বিস্তারিত হিসাবে, সামগ্রিক পরিমাণগত বিশ্লেষণ বিভিন্ন মূল পর্যবেক্ষণকে শক্তিশালী করে:

* **সেরা সমাধানে পৌঁছানোর ক্ষেত্রে সামঞ্জস্যতা:** HC\_MaxIter\_500 এবং GA\_Config\_1 ও GA\_Config\_4-এর অপ্টিমাইজ করা সংস্করণসহ বিভিন্ন কনফিগারেশন ধারাবাহিকভাবে **0.057143**-এর সেরা পরিচিত ফিটনেস অর্জনের ক্ষমতা প্রদর্শন করেছে।
* **হিল ক্লাইম্বিংয়ের উচ্চতর দক্ষতা:** HC\_MaxIter\_500 দ্রুততম অ্যালগরিদম হিসাবে আবির্ভূত হয়েছে, প্রায় **0.44 সেকেন্ডে** সর্বোত্তম সমাধান খুঁজে পেয়েছে, যা জেনেটিক অ্যালগরিদমগুলির চেয়ে উল্লেখযোগ্যভাবে দ্রুত।
* **অপ্টিমাইজড জেনেটিক অ্যালগরিদমগুলির কর্মক্ষমতা:** GA\_Config\_1\_mutation\_rate\_0.25 (প্রায় 6.25 সেকেন্ড) এবং GA\_Config\_4\_mutation\_rate\_0.15 (প্রায় 8.67 সেকেন্ড)-এর মতো কনফিগারেশনগুলিও নিখুঁত সামঞ্জস্যের সাথে সেরা ফিটনেস অর্জন করেছে, যদিও এতে বেশি সময় লেগেছে। এগুলি সমাধান স্থানের বিস্তৃত অন্বেষণ সরবরাহ করে।
* **প্যারামিটার অপ্টিমাইজেশানের প্রভাব:** GA-গুলির কর্মক্ষমতা পরিমার্জন করার জন্য প্যারামিটার পরিবর্তনের পর্যায়টি অপরিহার্য ছিল, যেখানে উচ্চতর মিউটেশন হার এবং বৃহত্তর জনসংখ্যা সাধারণত উন্নত সমাধানের গুণমান এবং সামঞ্জস্যের দিকে পরিচালিত করে।

### ৬.২. চূড়ান্ত সুপারিশসমূহ

সম্পূর্ণ কাজের এবং একত্রিত পরিমাণগত ফলাফলের উপর ভিত্তি করে:

১. **গতির অগ্রাধিকার সহ পরিস্থিতিগুলির জন্য:** যদি সম্পাদনের সময়টি গুরুত্বপূর্ণ বিষয় হয় এবং পরিচিত সর্বোত্তম সমাধান গ্রহণযোগ্য হয়, তবে **500 পুনরাবৃত্তি সহ হিল ক্লাইম্বিং (HC\_MaxIter\_500)** প্রাথমিক সুপারিশ, কারণ এর ব্যতিক্রমী গতি এবং ধারাবাহিকভাবে সেরা সমাধান খুঁজে বের করার ক্ষমতা রয়েছে। ২. **সর্বোচ্চ গুণমান এবং অন্বেষণ প্রয়োজন এমন পরিস্থিতিগুলির জন্য:** যদি সমাধান স্থানের আরও পুঙ্খানুপুঙ্খ অন্বেষণের প্রয়োজন হয়, সর্বোত্তম সম্ভাব্য ফিটনেস অর্জনের নিশ্চয়তা সহ, এবং সম্পাদনের সময় একটি গৌণ বিবেচনা হয়, তবে অপ্টিমাইজ করা জেনেটিক অ্যালগরিদম কনফিগারেশনগুলি предпочтительнее। বিশেষত, **0.15 মিউটেশন হার সহ GA\_Config\_4 এবং 0.25 মিউটেশন হার সহ GA\_Config\_1** সুপারিশ করা হয়, যা সমাধানের গুণমান এবং পরিচালনাযোগ্য সম্পাদনের সময়ের মধ্যে একটি ভাল ভারসাম্য সরবরাহ করে। ৩. **ভবিষ্যতের কাজ:** \* হিল ক্লাইম্বিংয়ের গতি এবং জেনেটিক অ্যালগরিদমগুলির অন্বেষণ ক্ষমতাকে একত্রিত করে এমন হাইব্রিড কৌশলগুলি অন্বেষণ করা। \* বিভিন্ন সমাধান উপস্থাপনা বা আরও জটিল ফিটনেস ফাংশনের প্রভাব তদন্ত করা। \* তাদের পরিমাপযোগ্যতা এবং সাধারণীকরণ মূল্যায়ন করার জন্য বৃহত্তর ডেটাসেট বা বিভিন্ন বৈশিষ্ট্যযুক্ত ডেটাসেটগুলিতে অপ্টিমাইজ করা অ্যালগরিদমগুলি প্রয়োগ করা। \* সিমুলেশন সেটআপ এবং সম্পাদন সহজতর করার জন্য আরও ব্যবহারকারী-বান্ধব ইন্টারফেস তৈরি করা।

এই প্রকল্পটি ক্রীড়া দল গঠনের সমস্যা সমাধানে মেটাহিউরিস্টিকগুলির কার্যকারিতা প্রদর্শন করেছে এবং একটি শক্তিশালী বিশ্লেষণ পাইপলাইন সরবরাহ করেছে, যার ফলাফল এবং প্রক্রিয়াগুলি প্রতিটি পর্যায়ের জন্য সরবরাহ করা নোটবুকগুলির মাধ্যমে পুঙ্খানুপুঙ্খভাবে নথিভুক্ত এবং পুনরুৎপাদনযোগ্য।

(পরিকল্পনা অনুযায়ী পরিশিষ্ট এবং বিশ্লেষণ প্রক্রিয়া ডায়াগ্রাম বিভাগ পরে যোগ/উল্লেখ করা হবে।)

## পরিশিষ্ট

এই বিভাগে CIFO এক্সটেন্ডেড প্রকল্পের সময় উৎপাদিত এবং বিশ্লেষণ করা বিস্তারিত সহায়ক নথি রয়েছে।

### পরিশিষ্ট ক: একক-প্রসেসর (এসপি) ধাপের নথি

* একক-প্রসেসর ধাপের রিপোর্ট: আর্কিটেকচার, অ্যালগরিদম এবং অপারেটর (reports\_sp\_phase/bn/01\_sp\_architecture\_algorithms\_operators.md)
* এসপি ধাপের আর্কিটেকচার ডকিউমেন্ট (reports\_sp\_phase/bn/00\_sp\_architecture.md)
* এসপি ধাপের কোড পর্যালোচনা ডকিউমেন্ট (reports\_sp\_phase/bn/00\_sp\_code\_review.md)

### পরিশিষ্ট খ: মাল্টিপ্রসেসিং (এমপি) ধাপের নথি

* এমপি ধাপের রিপোর্ট - ৫টি সম্পাদনের বিশ্লেষণ (reports\_mp\_phase/bn/02\_mp\_5\_runs\_analysis.md)
* এমপি ধাপের রিপোর্ট - ৩০টি সম্পাদনের পরিসংখ্যানগত বিশ্লেষণ (reports\_mp\_phase/bn/03\_mp\_30\_runs\_statistical\_analysis.md)
* এমপি ধাপের রিপোর্ট - প্যারামিটার পরিবর্তনের বিশ্লেষণ (reports\_mp\_phase/bn/04\_mp\_param\_var\_analysis.md)
* এমপি ধাপের রিপোর্ট - চূড়ান্ত প্যারামিটার পরিবর্তনের পরিসংখ্যানগত বিশ্লেষণ (reports\_mp\_phase/bn/05\_mp\_final\_param\_var\_statistical\_analysis.md)
* এমপি ধাপের আর্কিটেকচার ডকিউমেন্ট (reports\_mp\_phase/bn/00\_mp\_architecture.md)
* এমপি ধাপের কোড পর্যালোচনা ডকিউমেন্ট (reports\_mp\_phase/bn/00\_mp\_code\_review.md)

### পরিশিষ্ট গ: পরিমাণগত বিশ্লেষণ এবং ডেটা

* বিস্তারিত পরিমাণগত বিশ্লেষণ ডকিউমেন্ট (quantitative\_analysis/bn/quantitative\_analysis\_bn.md)
* ফলাফলের সারাংশ - ৫টি এমপি সম্পাদন (images\_mp/run\_5\_results/all\_algorithms\_summary\_mp\_5runs.csv)
* ফলাফলের সারাংশ - ৩০টি এমপি সম্পাদন (প্রতিশ্রুতিশীল) (images\_mp/run\_30\_results\_promising/all\_algorithms\_summary\_mp\_30runs.csv)
* ফলাফলের সারাংশ - জিএ প্যারামিটার পরিবর্তন (images\_mp/param\_var\_results/all\_ga\_variations\_summary\_mp\_5runs.csv)
* ফলাফলের সারাংশ - চূড়ান্ত প্যারামিটার পরিবর্তন (৩০টি সম্পাদন) (images\_mp/final\_param\_var\_results/all\_algorithms\_summary\_final\_param\_var\_30runs.csv)

### পরিশিষ্ট ঘ: বিশ্লেষণ নোটবুক (অবস্থান)

* বিশ্লেষণ এবং ফলাফল তৈরির জন্য ব্যবহৃত সমস্ত জুপিটার নোটবুক প্রকল্পের সংগ্রহস্থলের /notebooks/ ডিরেক্টরিতে অবস্থিত।
  + notebooks/01\_single\_processor\_analysis.ipynb (উদাহরণ, আপডেট/তৈরি করা হবে)
  + notebooks/02\_multiprocessing\_5\_runs\_analysis.ipynb (উদাহরণ, আপডেট/তৈরি করা হবে)
  + notebooks/03\_multiprocessing\_30\_runs\_analysis.ipynb (উদাহরণ, তৈরি করা হবে)
  + notebooks/04\_multiprocessing\_param\_var\_analysis.ipynb (উদাহরণ, তৈরি করা হবে)
  + notebooks/05\_multiprocessing\_final\_param\_var\_analysis.ipynb (উদাহরণ, তৈরি করা হবে)

### পরিশিষ্ট ঙ: বিশ্লেষণ প্রক্রিয়া চিত্র

* (বিশ্লেষণ প্রক্রিয়া চিত্রের ছবির রেফারেন্স, যা এখানে অন্তর্ভুক্ত বা উল্লেখ করা হবে)